```c
#include <stdio.h>

int main() {
    int n, i, j, temp;
    int bt[20], p[20], wt[20], tat[20];
    float avg_wt = 0, avg_tat = 0;

    printf("Enter total number of processes: ");
    scanf("%d", &n);

    printf("Enter burst time for each process:\n");
    for (i = 0; i < n; i++) {
        printf("P[%d]: ", i + 1);
        scanf("%d", &bt[i]);
        p[i] = i + 1; // Store process number
    }

    // Sort processes by burst time (SJF)
    for (i = 0; i < n - 1; i++) {
        for (j = i + 1; j < n; j++) {
            if (bt[i] > bt[j]) {
                temp = bt[i];
                bt[i] = bt[j];
                bt[j] = temp;

                temp = p[i];
                p[i] = p[j];
                p[j] = temp;
```

```
        }
    }
}

wt[0] = 0; // First process has 0 waiting time

// Calculate waiting time
for (i = 1; i < n; i++) {
    wt[i] = 0;
    for (j = 0; j < i; j++)
        wt[i] += bt[j];
}

// Calculate turnaround time
for (i = 0; i < n; i++) {
    tat[i] = bt[i] + wt[i];
    avg_wt += wt[i];
    avg_tat += tat[i];
}

printf("\nProcess\tBurst Time\tWaiting Time\tTurnaround Time\n");
for (i = 0; i < n; i++) {
    printf("P[%d]\t%d\t\t%d\t\t%d\n", p[i], bt[i], wt[i], tat[i]);
}

avg_wt /= n;
avg_tat /= n;
```

```
    printf("\nAverage Waiting Time = %.2f", avg_wt);
    printf("\nAverage Turnaround Time = %.2f\n", avg_tat);

    return 0;
}
```

```
Enter total number of processes: 4
Enter burst time for each process:
P[1]: 6
P[2]: 7
P[3]: 8
P[4]: 3

Process Burst Time      Waiting Time    Turnaround Time
P[4]    3               0               3
P[1]    6               3               9
P[2]    7               9               16
P[3]    8               16              24

Average Waiting Time = 7.00
Average Turnaround Time = 13.00


...Program finished with exit code 0
Press ENTER to exit console.
```